

Tartu Ülikool
Loodus- ja täppiseaduste valdkond
Matemaatika ja statistika instituut

Mihkel Lepson
Rekurrentsed närvivõrgud aegridade uurimiseks
Matemaatilise statistika eriala
Bakalaureusetöö (9 EAP)

Juhendaja: PhD. Raivo Kolde

TARTU 2020

Rekurrentsed närvivõrgud aegridade uurimiseks

Bakalaureusetöö

Mihkel Lepson

Lühikokkuvõte. Bakalaureusetöö eesmärk on välja selgitada, milliseid seoseid suudab aegreast ülesleida erilist tüüpi rekurrentne närvivõrk LSTM, et selle abil teha ennustusi tuleviku väärtuste kohta. Püstitatud eesmärgi uurimiseks vaadatakse kõigepealt LSTM mudeli võimekust leida seoseid genereeritud andmete peal. Genereeritud andmestikus on olemas kindlat tüüpi seosed, mida soovitakse mudeliga leida. Seosteks on sündmustest põhjustatud mõju mõõdetavale tunnusele, süstemaatiline muutus mõõdetavas tunnuses, tsükliline muutus mõõdetavas tunnuses. Kui mudel suudab leida genereeritud andmete pealt seosed ja ennustada tuleviku väärtusi, siis peaks suutma ka LSTM mudel seda teha päris andmestiku peal. Päris andmestikuks valitakse meditsiini andmestik, mis on pärit MIMIC-III andmestikust. Vaadatakse täpsemalt LSTM mudeli võimekust ennustada viite elulist näitajat, milleks on südamerütm, hingamissagedus, hapniku küllastus veres, diastoolne vererõhk ja süstoolne vererõhk.

CERCS teaduseriala: P160 Statistika, operatsioonanalüüs, programmeerimine, finants- ja kindlustusmatemaatika.

Märkõnad: aegridade analüüs, masinõpe, tehisnärvivõrgud.

Recurrent neural networks for time series analysis

Bachelor thesis

Mihkel Lepson

Abstract. The objective of this bachelor's thesis is to find out which type of relations can recurrent neural network LSTM find from time series to forecast future values. First, the LSTM model's ability to find relations and forecast future values will be examined

on the generated data. In the generated data there will be specific type of relations like events effect on the measurement, the systematic change in the measurement and the cyclic change in the measurement. If LSTM model is able to find relations on generated data then it's ability to find relations and forecast future values be examined on the real data. The medical data from MIMIC-III database will be selected as the real data. The LSTM model's ability will be examined on five vital signs which are heart rate, respiratory rate, oxygen saturation in blood, diastolic blood pressure and systolic blood pressure.

CERCS research specialisation: P160 Statistics, operation research, programming, actuarial mathematics.

Key Words: time series, machine learning, artificial neural networks .

Sisukord

Sissejuhatus	6
2 Tehisnärvivõrgud	7
2.1 LSTM	8
2.1.1 LSTM struktuur	8
2.2 Väljajätumeetod	10
2.3 Kaofunktsioon	10
2.4 Närvivõrgu optimeerimine	11
2.4.1 Järsima laskumise meetod	11
2.4.2 Inertsiga järsima laskumise meetod	11
2.4.3 <i>Root Mean Square Propagation</i>	12
2.4.4 Optimeerimismeetod Adam	12
2.5 Närvivõrgu treenimine	13
3 LSTM mudel genereeritud andmetel	15
3.1 Andmete genereerimine	15
3.2 Mudelite ehitus, treenimine ja tulemused	17
3.2.1 Esimese genereeritud andmestiku mudel	17
3.2.2 Teise genereeritud andmestiku mudel	18
3.2.3 Kolmanda genereeritud andmestiku mudel	20
3.2.4 Neljanda genereeritud andmestiku mudel	21
3.3 Järeldused	22

4	LSTM mudel meditsiini andmetel	23
4.1	Andmestiku ülevaade	23
4.2	Andmete jaotamine	25
4.3	Mudelite ehitus ja treenimine	25
4.4	Mudelite tulemused ja võrdlus	27
4.5	Mudelite järelused	32
	Kokkuvõtte	34
	Kasutatud kirjandus	35
	Lisad	36
	Lisa 1, Genereeritud andmed	36
	Lisa 2, Mudelite treenimine genereeritud andmetel	38
	Lisa 3, Mudelite treenimine meditsiini andmetel	40

Sissejuhatus

Ennustus mudelite tegemiseks on mitmeid erinevaid võimalusi ning üks võimalus teha ennustus-mudelit on kasutada tehishärvivõrke. Leidub erinevaid tehishärvivõrgu struktuure, aga käesolevas töös keskendutakse LSTM härvivõrkudele ehk kindlat tüüpi rekurrentsele härvivõrgule. Eesmärk on vaadata, kas LSTM mudelid suudavad leida aegreast üles teatud tüüpi seosed ja selle põhjal teha ennustusi tuleviku väärtuste kohta.

Püstitatud küsimuse uurimiseks võetakse meditsiini andmestik MIMIC-III, kus on mõõdetud aegread ning erinevad tunnused nagu vanus ja manustatud ravimid. Mõõdetud on täpsemalt viiete järgnevat elulist näitajat: südamerütm, hingamissagedus, hapniku küllastus veres, diastoolne vererõhk ja süstoolne vererõhk. Teada saamaks, kas LSTM mudel suudab leida aegreast üles ravimite põhjustatud mõjud mõõdetavale tunnusele või näiteks eelnevatest väärtustest põhjustatud seoseid, siis genereeritakse sarnased andmestikud, kus on olemas kindlad seosed. Genereeritud andmestikus on olemas järgnevad kolm seost mida vaadatakse: sündmuse mõju mõõdetavale tunnusele, süstemaatiline muutus mõõdetavas tunnuses ja tsükliline ehk kellaajast põhjustatud mõju. Kui LSTM mudel leiab genereeritud andmete peal seosed üles ja suudab nende põhjal teha ennustusi tuleviku väärtuste kohta, siis peaks LSTM mudel suutma seda teha ka meditsiini andmetel.

Hüpoteesiks seatakse, et LSTM mudel suudab leida genereeritud andmete pealt üles kõik vajalikud seosed ja nende põhjal ennustada tuleviku väärtusi. Teine hüpotees on see, et meditsiini andmete pealt suudab LSTM mudel samuti üles leida seosed ja tuleviku väärtusi ennustada. See eest mitte nii täpselt nagu genereeritud andmete peal.

Töös on mudelid tehtud R'i paketiga Keras. Mudelid on arvutatud Tartu Ülikooli Teadusarvutuste keskuse ressurssidega.

2 Tehisnärvivõrgud

Tehisnärvivõrk on omavahel ühendatud sõlmede kogum, kus iga sõlm saab mööda ühendusi sisendi ja annab ka väljundi. Sõlme sisendid saab kirja panna vektorina $X^{(i)} = [x_1, \dots, x_{m_i}]$, kus $x_l \in \mathbb{R}$, $l \in \{1, \dots, m_i\}$. Igal ühendusel on arvuline kaal $w_{l,j} \in \mathbb{R}$, kus l tähistab lähte sõlme ja j tähistab siht sõlme. Sisendite kaalud saab kirja panna vektorina $W_j^{(i)} = [w_{1,j}, \dots, w_{m_{i-1},j}]$. Siit saame, et i -nda kihi j -nda sõlme sisendiks on

$$s_j^{(i)} = b_j^{(i)} + X^{(i-1)} \cdot W_j^{(i)T} = \sum_{l=0}^{m_{i-1}} x_l \cdot w_{l,j}$$

, kus b on vabaliige (Callan, 2003, lk 286-290). Saadud tulemusele rakendatakse ka aktivatsioonifunktsiooni, et tegemist ei oleks ainult lineaarse teisendusega (Russel & Norvig 2016: 729). Aktivatsioonifunktsioone on olemas mitmeid, kuid antud töös vaadatakse täpsemalt kahte: sigmoidfunktsiooni ja hüperboolset tangensfunktsiooni.

Sigmoidfunktsioonidest vaadatakse täpsemalt logistilist funktsiooni, mis avaldub valemiga

$$\sigma(s_j^{(i)}) = \frac{1}{1 + e^{-s_j^{(i)}}}$$

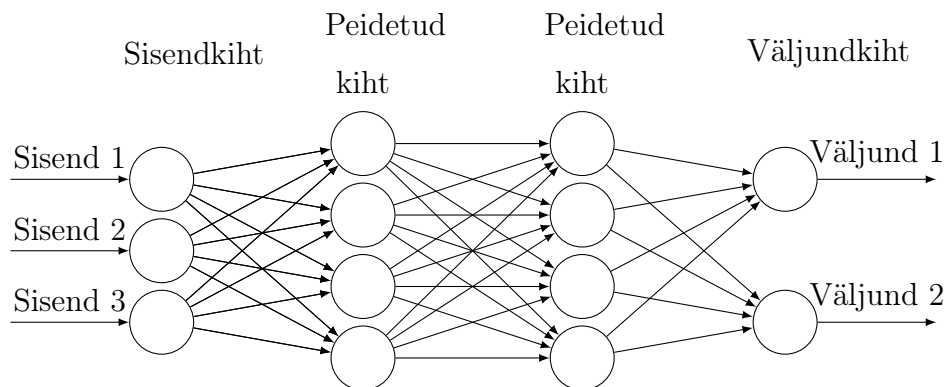
Kuna $e^{-s_j^{(i)}} \in (0, \infty)$, siis funktsiooni väärtused jäävad vahemikku nullist üheni. Hüperboolne tangensfunktsioon avaldub valemiga

$$\tanh(s_j^{(i)}) = \frac{e^{2s_j^{(i)}} - 1}{e^{2s_j^{(i)}} + 1}$$

Kuna $e^{2s_j^{(i)}} \in (0, \infty)$, siis funktsiooni väärtused jäävad vahemikku miinus ühest üheni.

Närvivõrgus on sõlmed kihtides nagu on näha jooniselt 1. Kiht on sõlmede kogum, kus kõik sõlmed saavad sama sisendi. Enne olid kihid tähistatud ülaindeksiga $^{(i)}$, kus $i \in \{0, 1, 2, \dots, k\}$. $i = 0$ on sisendkiht, $i \in \{1, 2, \dots, k-1\}$ on peidetud kihid ja $i = k$ on väljundkiht. Nimetus peidetud kiht tuleneb sellest, et kihis olevad sõlmed ei saa väärtusi otse andmestikust ega tagasta ühtegi sõlme tulemust kasutajale (Callan, 2003, lk 292). Sõlmede ühendamisel on kaks võimalust. Üks võimalus on teha pärilevivõrk ja teine on teha rekurrentne võrk (Russel & Norvig, 2016, lk 729). Rekurrentse närvivõrgu erinevus tavalise pärilevi võrguga on see, et rekurrentne närvivõrk

jätab meelde ka varasemad mudeli väärtused ja kasutab seda informatsiooni tulevaste sisendite korral (Russel & Norvig, 2016, lk 729). See annab võimaluse aegridade korral leida täpsemaid tuleviku väärtusi.



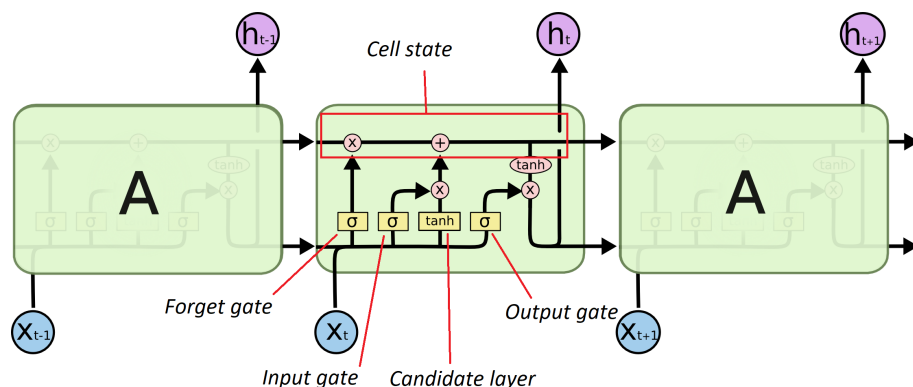
Joonis 1: Närvivõrgu struktuur

2.1 LSTM

Long short term memory, edaspidi LSTM, on erist tüüpi rekurrentne närvivõrk, mille eesmärk on leida seoseid aegreas. LSTM erinevus tavalise rekurrentse närvivõrguga seisneb selles, et LSTM leiab andmestikust ka pikaajalised seosed (Olah, 2015). Kui patsiendile manustatakse ravimit, mille toime mõõdetavale tunnusele on pikaajalisem või ravimi toime avaldub alles mitme tunni pärast, siis erinevalt tavalisest rekurrentsest närvivõrgust peaks LSTM need seosed paremini leidma.

2.1.1 LSTM struktuur

Käesolev alapeatükk põhineb allikal (Olah, 2015). LSTM sõlme struktuuri on näha jooniselt 2 (Olah, 2015) (Joonist on täiendatud autori poolt). Põhiline idee LSTM struktuuri taga on üksusel *cell state*. *Cell state*, mida võib nimetada ka mäluks. *Cell state* kannab informatsiooni



Joonis 2: LSTM struktuur

ühelt ajahetkelt teisele. Veel koosneb LSTM neljast erinevast osast: *forget gate*, *candidate layer*, *input gate* ja *output gate*.

Forget gate'i eesmärk on otsustada, kui palju *cell state*'is olevat informatsiooni säilitada. *Forget gate* saab ajahetkel t sisendiks vektori x_t ja eelmise ajahetke LSTM'i väljundi vektori h_{t-1} . Saadud sisendid korrutatakse läbi kaalude vektoriga W . Korrutise tulemusele liidetakse vabaliige b ja saadud summale rakendatakse sigmoidfunktsiooni.

$$f_t = \sigma([h_{t-1}, x_t] \cdot W_f^T + b_f)$$

Kuna rakendatakse sigmoidfunktsiooni, siis *forget gate* väljundi väärtused jäävad vahemikku nullist üheni. Mida ühe lähedasem on väärtus seda olulisem on *cell state*'is olev väärtus.

Input gate'i ja *candidate layer*'i eesmärk on *cell state*'is oleva informatsiooni uuendamine. Mõlemad saavad sama sisendi mis *forget gate*. *Input gate*'ile rakendatakse sigmoidfunktsiooni ja *candidate layer*'ile hüperboolset tangensfunktsiooni.

$$i_t = \sigma([h_{t-1}, x_t] \cdot W_i^T + b_i)$$

$$\tilde{C}_t = \tanh([h_{t-1}, x_t] \cdot W_C^T + b_C)$$

Cell state uuendamiseks leitakse vana *cell state*'i korrutis *forget gate*'ga ja tulemusele liidetakse *input gate*'i ja *candidate layer*'i korrutis.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

Output gate’i eesmärk on filtreerida *cell state* väljundi. *Output gate* saab sama sisendi, mis *forget gate* ja samuti rakendatakse tulemusele sigmoidfunktsiooni.

$$o_t = \sigma([h_{t-1}, x_t] \cdot W_o^T + b_o)$$

Sõlmest väljundi saamiseks rakendatakse *cell state*’le hüperboolset tangensfunktsiooni, et väärtused jääksid miinus ühest üheni. Seejärel leitakse saadud tulemuse korrutis *output gate*’ga. Korrutise tulemus on ajahetke t väljund.

$$h_t = o_t \cdot \tanh(C_t)$$

2.2 Väljajätumeetod

Väljajätumeetod on üks levinumaid viise kuidas regulariseerida närvivõrku. Väljajätumeetodi rakendatakse tavaliselt närvivõrgu peidetudkihi väljundile ja rakendatakse ainult mudeli treenimise käigus, et võidelda ülesobitamise vastu. Idee seisneb selles, et vastavalt väljajätumeetodi määrale, mis on tavaliselt lõigus 0,2 kuni 0,5, muudetakse juhuslikult valitud väljundid nulliks. Valideerimise käigus seda ei rakendata, vaid peidetudkihi väärtuseid vähendatakse $1 - (\text{väljajätumäär})$ kordi (Chollet ja Allaire, 2018).

2.3 Kaofunktsioon

Närvivõrgustiku kaalude muutmiseks on vaja mõõta, kui palju närvivõrgustikuga saadud tulemus erineb tegelikkusest. Selle jaoks võetakse kasutusele kaofunktsioon. Kaofunktsioon saab sisendiks närvivõrguga saadud tulemuse ja tegeliku tulemuse ning tagastab arvulise väärtuse ehk kao (Chollet ja Allaire, 2018). Antud töös vaadatakse pidevaid tunnuseid. Pidevate tunnuste korral sobib hästi kaofunktsiooniks keskmine ruutviga (Abrahamson, 2018). Keskmine ruutviga avaldub valemiga

$$\frac{1}{n} \sum_{i=1}^n (h_i - \hat{h}_i)^2$$

kus h_i on vaatluse i soovitud tulemus ja \hat{h}_i on mudeliga saadud tulemus.

2.4 Närvivõrgu optimiseerimine

2.4.1 Järsima laskumise meetod

Närvivõrgu optimiseerimiseks nimetatakse optimaalsete parameetrite leidmist, mille korral kaofunktsioon on minimaalne. Teisisõnu otsitakse kaofunktsiooni globaalset miinimumi. Kaofunktsiooni puhul on veaks

$$\varepsilon = \frac{1}{n} \sum_1^n (h_i - g(x_i, \theta))^2,$$

kus $g(x_i, \theta) = \hat{h}_i$ ja parameeter θ on vektor, mis koosneb vektorite W_f, W_i, W_C, W_o elementidest ja vabaliikmedest b_f, b_i, b_C, b_o . Vea gradient avaldub valemiga

$$\nabla_{\theta} f_t(\theta_{t-1}) = \frac{\partial \varepsilon}{\partial \theta_{t-1}}$$

Uuendatud parameetriteks saadakse

$$\theta_t \leftarrow \theta_{t-1} - \alpha \nabla_{\theta} f_t(\theta_{t-1}),$$

kus α on õpisammu suurus (Nilsson, 1998). Õpisamm ütleb seda, kui palju parameetreid antigradiendi suunas muuta. Kui õpisamm on liiga suur, siis koondumist globaalsesse miinimumi ei toimu. Kui õpisamm on liiga väike, siis koondumine globaalsesse miinimumi on liiga aeglane (Murphy 2012).

2.4.2 Inertsiga järsima laskumise meetod

Käesolev alapeatükk põhineb allikal (Rizwan, 2018). Idee inertsiga järsima laskumise meetodi juures on kiirendada koondumist globaalsesse miinimumi. Sellisel juhul võtab meetod piltikult otsesema tee globaalse miinimumi suunas. Parameetrite uuendamine on järgnev

$$m_t \leftarrow \beta m_{t-1} + (1 - \beta) \nabla_{\theta} f_t(\theta_{t-1}),$$

kus $m_0 = 0$ ja β mis jääb vahemikku nullist üheni on inerts. Uuendatud parameetriteks on

$$\theta_t \leftarrow \theta_{t-1} - \alpha m_t$$

Erinevalt tavalisest järsima laskumise meetodist võtab antud meetod arvesse ka varasemaid gradiente.

2.4.3 Root Mean Square Propagation

Käesolev alapeatükk põhineb allikal (Kathuria, 2018). *Root Mean Square Propagation*, edaspidi RMSprop, on järsima laskumise meetod. Erinevus võrreldes varasemate meetoditega seisneb selles, et RMSprop meetodis on õpisamm igal ajahetkel erinev. Muutes õpisammu suurst, jõuab antud meetod kiiremini globaalsesse miinimumi.

$$v_t \leftarrow \beta v_{t-1} + (1 - \beta)(\nabla_{\theta} f_t(\theta_{t-1}))^2,$$

kus $v_0 = 0$. Nagu näha, siis valem on sarnane inertsiga järsima laskumise meetodiga, kuid erinevus tuleneb sellest, et üks korrutis on gradienti ruut ehk vektori hadamardi korrutis iseendaga. Uuendatud parameetriteks saadakse

$$\theta_t \leftarrow \theta_{t-1} - \alpha \frac{\nabla_{\theta} f_t(\theta_{t-1})}{\sqrt{v_t + \epsilon}},$$

kus ϵ on nulli lähedane arv. Viimast on vaja liita selleks, kuna võib juhtuda olukord, kus v_t on võrdne nulliga.

2.4.4 Optimeerimismeetod Adam

Käesoleva meetodi kirjeldus põhineb artiklil “Adam: A method for stochastic optimization” (Kingma ja Ba, 2017). Adam on optimeerimisalgoritm parameetrite uuendamiseks närvivõrgus ja kasutab ära mõlema eelneva meetodi häid omadusi. Adam on robustne ja töötab hästi ka mittestatsionaarsel andmestikul. Adam rakendamiseks on vaja määrata neli parameetrit ja kaofunktsioon. Kaofunktsiooniks määratakse keskmine ruutviga tähisega f .

$$f(\theta) = \frac{1}{n} \sum_{i=1}^n (h_i - g(x_i, \theta))^2 \quad (1)$$

Eesmärk on valemis (1) muuta mudeli g parameetrit θ niiviisi, et f väärtus oleks võimalikult väike ehk minimiseerida f oodatavat väärtust $E[f]$. Veel on vaja määrata α ehk õpisammu suurus, $\beta_1, \beta_2 \in [0, 1)$ ehk momentide hinnangute eksponentsiaalsed kahanemiskiirused. Lõpuks on vaja määrata ϵ , mis on nullilähedane arv. Kõikide parameetrite väärtusteks valitakse meetodi autorite

pakutud universaalsed väärtused.

$$\alpha = 0,001$$

$$\beta_1 = 0,9$$

$$\beta_2 = 0,999$$

$$\epsilon = 10^{-8}$$

Adam algoritm parameetrite uuendamiseks on järgmine. Esmalt arvutatakse gradient ajahetkel t .

$$g_t = \nabla_{\theta} f_t(\theta_{t-1})$$

Seejärel uuendatakse nihkega esimese momendi hinnangut ja nihkega teist kohandamata momenti.

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$$

$$v_t = \beta_2 \cdot m_{t-1} + (1 - \beta_2) \cdot (g_t \circ g_t)$$

Mõlemale leitud väärtusele tehakse nihke parandus.

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

$$\hat{v}_t = \frac{b_t}{1 - \beta_2^t}$$

Uuendatud parameetrite väärtuseks saadakse

$$\theta_t = \theta_{t-1} - \frac{\alpha \cdot \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

2.5 Närvivõrgu treenimine

Kõik mudelid antud töös töötavad samamoodi. Mudel saab sisendiks esimese ajahetke tunnused ja selle põhjal proovib ennustada teise ajahetke mõõtmiste tulemusi. Seejärel saab mudel sisendiks teise ajahetke tegelikud mõõtmistulemused ja üritab ennustada kolmanda ajahetke mõõtmiste tulemusi jne. Kui vaadata joonist 2, siis sisend vektoris X_{t-1} on olemas element, mis on mõõdetud

tunnuse väärtus ajahetkel $t - 1$. Väärtus h_{t-1} , mida mudel ajahetkel $t - 1$ väljastab, on mudeli ennustus selle kohta, mis on X_t üks element ehk ajahetke t mõõdetud tunnuse väärtus.

Kõikide mudelite treenimine töötab sarnaselt. Mudelitele ei anta korraga kogu treeningandmestiku ette, vaid mudelid saavad külastusi 64tes plokkides. Andes ette korraga andmestikust väiksemaid plokke, mitte kogu andmestiku, suudab optimiseerimisalgoritm leida täpsemad parameetrid (Luschi & Masters, 2018). Võimalik on anda ette ka 64st väiksemaid plokke, aga siis läheb mudelite treenimine liiga ajakulukaks. Samuti tehakse iga mudeliga mitu epohhi. Epohhiks nimetatakse ühte iteratsiooni, mille korral käiakse treeningandmestik läbi. Tehes mitmeid epohhe, leitakse optimaalsemad parameetrid mudelile (Chollet ja Allaire, 2018).

Teadsaamiseks täpset epohhide arvu, mille korral mudel saavutab optimaalsed parameetrid, tehakse iga mudeliga esialgu 300 epohhi. Kui mudel hakkab selle ajajooksul ületreenima, siis treenitakse mudelid uuesti, tehes vähem epohhe. Ületreenimine tähendab seda, et lisades epohhe juurde, mudeli kadu suureneb. See on tingitud sellest, et mudel leiab valimi põhised seosed, mis ei kehti üldkogumis (Chollet ja Allaire, 2018). Kui mudel ei hakanud ületreenima, siis lisatakse 100 või 200 epohhi, kuni leitakse optimaalsed parameetrid. Ennem 100 või 200 epohhi lisamist salvestatakse alati eelneva mudeli parameetrid, kuna võib juhtuda olukord, et optimaalsed parameetrid saavutatakse näiteks 450 epohhi korral. Sellisel juhul saab lisada 400 epohhiga saadud mudelile 50 epohhi juurde ja ei pea mudelit 450 epohhiga uuesti treenima.

3 LSTM mudel genereeritud andmetel

Vaatamaks LSTM mudeli võimekust leida aegreast seoseid, genereeritakse andmestikud, kus on olemas kindlad seosed. Vaadatakse mudeli võimekust leida sündmuse mõju mõõdetavale tunnusele, mudeli võimekust leida süstemaatilist muutust ja võimekust leida tsüklilist muutust.

3.1 Andmete genereerimine

Genereeritakse andmestik, et vaadata, kas LSTM leiab sündmuste mõjud mõõdetavale tunnusele. Andmestikus on üks mõõdetud tunnus ja kolm sündmust. Mõõdetud tunnus on vahemikus nullist üheni. Esimese sündmuse efekt avaldub järgneval kahel tunnil peale toimumist. Teise sündmuse efekt avaldub ülejärgmisel tunnil ja kestab ainult ühe tunni. Kolmanda sündmuse efekt avaldub kolm tundi peale toimumist ja kestab kogu ülejäänud mõõtmisperioodi. Esimene sündmus tõstab mõõdetud tunnust 0,1 võrra, teine sündmus alandab mõõdetud tunnust 0,05 võrra ja kolmas sündmus alandab mõõdetud tunnust 0,15 võrra. Kui kahel sündmusel avaldub efekt samal ajahetkel, siis nende mõjud liidetakse. Andmete genereerimine käib järgnevalt. Esimese ajahetke mõõdetud tunnused genereeritakse juhuslikult ühtlasest jaotusest vahemikus 0,3 kuni 0,5. Seejärel genereeritakse järgnevate mõõtmiste tulemused liites eelnevale tulemusele juhuslik suurus normaaljaotusest keskväärtusega null ja standardhälbega 0,005, kuni saadakse aegread pikkusega 24. Igale aegreale genereeritakse juurde sündmused. Esimese sündmuse tõenäosus toimuda igal ajahetkel aegreas on 0,1. Tõenäosus, et esimene sündmus toimub kahel ajahetkel järjest on null. Teise sündmuse tõenäosus toimuda igal ajahetkel on samuti 0,1. Kolmanda sündmuse tõenäosus toimuda igal ajahetkel on 0,05. Kuna sündmuse mõju kestab kogu ajaperioodi, siis võetakse sündmuse mõju alates esimesest toimumisest ja hilisemad toimumised ei avalda lisa mõju. Kui sündmused on genereeritud, siis lisatakse nende mõjud mõõdetud tulemustele. Saadakse järgnev andmestik lisa 1 joonis 15.

Teise genereeritud andmestiku eesmärk on vaadata, kas LSTM mudel suudab leida süstemaatilist muutust andmestikust. Selleks genereeritakse esimene arv aegreas ühtlasest jaotusest vahemikus 0,3 kuni 0,5. Teine ja kolmas väärtus saadakse eelnevale väärtusele liites juhuslik suurus nor-

maaljaotusest keskväärtusega 0 ja standardhälbega 0,01. Neljas väärtus saadakse kolmandale väärtusele liites juhuslik suurus normaaljaotusest, kuid see eest sõltub ka neljas väärtus kolmanda ja esimese väärtuse vahelisest seosest. Kui esimese ja kolmanda väärtuse vahe absoluutväärtus on väiksem kui 0,015, siis ei tehta neljanda väärtusega midagi. Kui kolmanda ja esimese väärtuse vahe on suurem kui 0,015, siis lahutatakse neljandast väärtusest 0,2 ja kui vahe on väiksem kui $-0,015$, siis liidetakse neljandale väärtusele 0,2. Järgnevad väärtused genereeritakse sarnaselt nagu eelnevad neli, kuni koos on 24 väärtust. Neljanda ja viienda väärtuse vahel ei ole seost. Samuti ei ole seost kaheksanda ja üheksanda väärtuse vahel jne. Saadakse järgnev andmestik lisa 1 joonis 16.

Kolmas andmestik on tsüklilise muutusega. Mõõdetud tunnuse väärtus sõltub kellaaajast. Selleks genereeritakse nagu esimesel juhul 10000 mõõtmiste perioodi pikkusega 24. Samuti genereeritakse juhuslikult igale mõõtmisele algus kellaaeg. Saadakse andmestik, kus üks mõõde on mõõdetud tunnus ja teine mõõde on mõõtmise kellaaeg. Mõõdetud tunnusele lisatakse ka kellaaaja mõju. Kui kellaaeg on 11 või 12, siis liidetakse mõõdetud tunnusele juurde suurus 0,06. Kui kellaaeg on 10 või 13, siis liidetakse juurde suurus 0,05. Minnes niiviisi edasi saadakse, et juhul kui kellaaeg on 5 või 18, siis ei liideta midagi juurde ning kui kellaaeg on 0 või 23, siis lahutatakse $-0,05$. Viimasena eemaldatakse andmestikust kellaaeg, et vaadata kas mudel suudab sellisel juhul seose leida. Saadakse järgnev andmestik lisa 1 joonis 17.

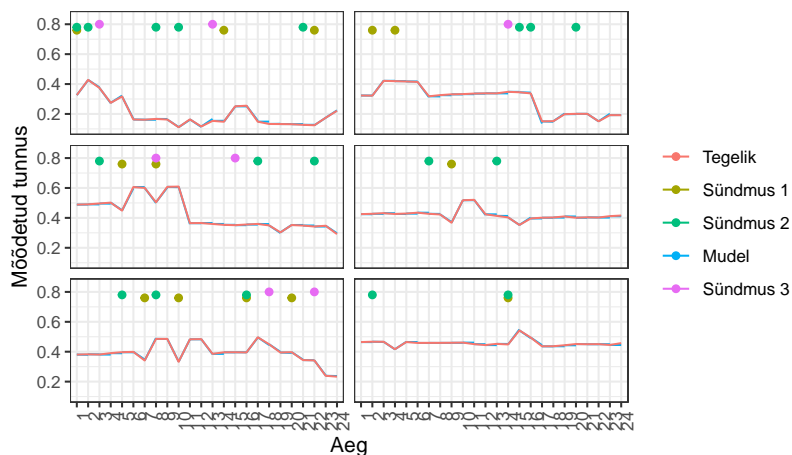
Neljandas andmestikus kombineeritakse kõik varasemad kolm andmestiku omadust. Aluseks valitakse teine andmestik. Sellele genereeritakse juurde esimese andmestiku puhul sündmused ja samuti lisatakse kellaaeg. Seejärel rakendatakse sündmuste mõju ja kellaaaja mõju. Peale kellaaaja mõju lisamist kellaaeg eemaldatakse. Võis juhtuda ka olukord, et mõni väärtus on väiksem nullist või suurem ühest. Kuna andmete genereerimise käigus on mõnede mõõtmiste väärtused väiksemad nullist, siis liidetakse igale väärtusele kõige väiksema väärtuse absoluutväärtus ja väärtused jagatakse uue suurima väärtusega. Antud tegevuse järjel on kõik väärtused vahemikus nullist üheni. Saadakse järgnev andmestik lisa 1 joonis 18.

3.2 Mudelite ehitus, treenimine ja tulemused

Mudelite tegemiseks jaotatakse kõik genereeritud andmestikud kaheks: treeningandmestik ja valideerimisandmestik. Treeningandmestikku võetakse esimesed 8000 mõõtmisperioodi ja valideerimisandmestikku 2000 viimast mõõtmisperioodi. Mõlemad andmestikud jaotatakse kaheks: sisendandmestik ja tulemusandmestik. Sisendandmestiku moodustavad esimesed 23 ajahetke ja tulemusandmestiku moodustavad viimased 23 ajahetke.

3.2.1 Esimese genereeritud andmestiku mudel

Esimesele genereeritud andmestikule konstrueeritakse lihtne mudel. Mudel koosneb kolmest kihist. Sisendkihi suurus on neli. Teises kihis on 32 LSTM sõlme ja väljundkihis on üks lihtne sõlm. Lihtsas sõlmes ei rakendata sõlme väljundile aktivatsioonifunktsiooni ning tegemist ei ole ka rekurrentse sõlmega. Mudelit treenitakse kuni saavutatakse optimaalsed parameetrid. Optimaalsed parameetrid saavutatakse 300 epohhi korral (lisa 2 joonis 19), mille puhul on mudeli kaoks 0,000026. Vaadates mudeli jääke, siis kõige väiksem jääk mudeliga on $-0,0231$ ja suurim 0,0186. Mõlemad jäägid on üle kahe korra väiksemad kui sündmuste mõjud mõõdetavale tunnusele, mille põhjal võib arvata, et iga aegrea puhul mudel leiab kõik seosed üles.

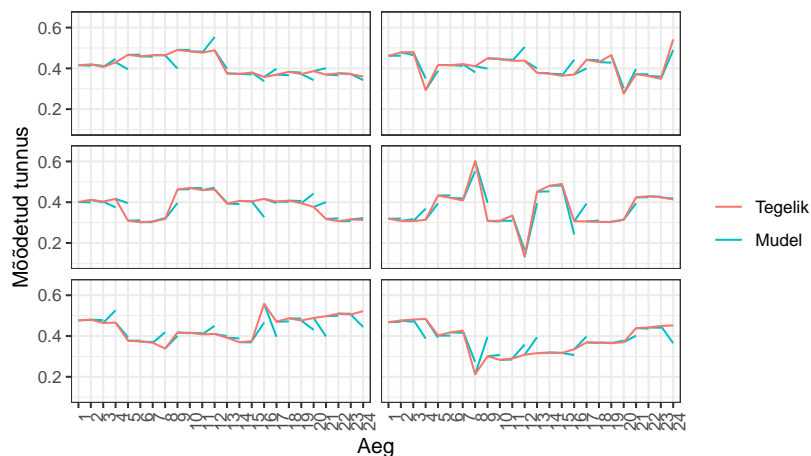


Joonis 3: Esimese genereeritud andmete ennustus

Nägemaks kuidas mudel ennustab, võetakse juhuslikult kuus mõõtmisperioodi ja selle ennustused. Vaadates joonist 3 on näha, et mudel ennustab kõik sündmustest tingitud muudatused mõõtmistes ära. Samuti suudab mudel ennustada ära ka mitme sündmuse koosmõju ning mudel on ka aru saanud, et kolmandal sündmusel on ainult ühekordne efekt ehk kui kolmandat sündmust on toimunud kaks korda, siis jääb mõõdetud tulemusele ainult ühekordne efekt. Järelikult suudab LSTM leida aegreast ära erinevat tüüpi sündmuste mõjud, millel on ka mõõdetavale tunnusele efekt.

3.2.2 Teise genereeritud andmestiku mudel

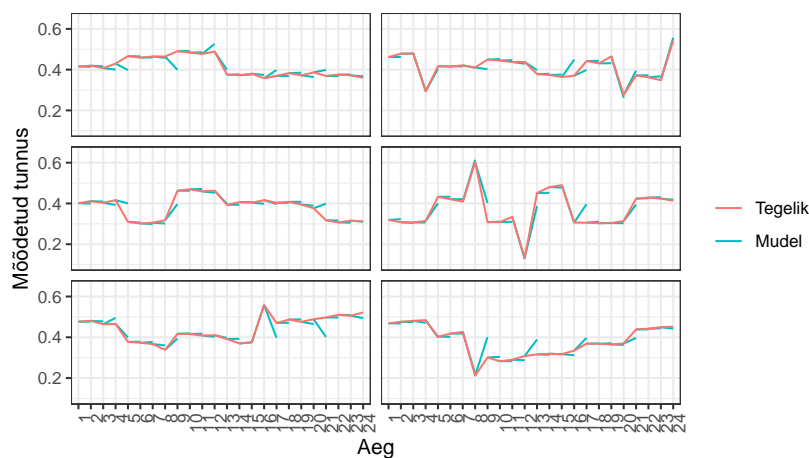
Teisele genereeritud andmestikule tehakse sama ehitusega mudel nagu esimesele andmestikule. Erinevus seisneb ainult sellest, et sisendkihi suurus on üks. Optimaalsed parameetrid saadakse 600 epohhi korral, mille puhul on mudeli kaoks 0,001502. Kadu on märksa suurem kui esimesel genereeritud andmestikul, kuna iga neljas väärtus alates esimesest on andmestikus juhuslik, mida on võimatu ette ennustada. Vaadates mudeli jääke saadakse, et väikseim jääk on $-0,1362$ ja suurim $0,1235$. Mõlema jäägi väärtus on suur, kuid mõlema absoluutväärtus on siiski väiksem kui $0,2$ ehk võib eeldada, et mudel leidis andmestikust seose, mille korral väärtused tõusevad ja langevad. Jääkide arvutamisel on aegreast iga neljas väärtus välja jäetud.



Joonis 4: Teise genereeritud andmete ennustus

Jooniselt 4 on näha, et mudel üldiselt ennustab hästi. Mudel on ka aru saanud, et iga neljas vaatlus on juhuslik ja ennustab, et iga neljas väärtus on 0,4, mis on ka ühtlase jaotuse keskväärtus vahemikus 0,3 kuni 0,5. See eest on näha, et mõnes kohas ennustab mudel vahest kummalist tõusu või langust. Tegemist ei ole ka süstemaatilise muutuse tõusu või langusega, kuna mudel ennustab, et väärtus tõuseb või langeb 0,1 võrra, aga süstemaatilise muutuse korral väärtus tõuseb või langeb 0,2 võrra.

Vaadatakse, kas suurem mudel suudab teha täpsemad ennustused. Uues mudeli ehituses on võrreldes varasemaga kaks lisa kiht. Juurde on lisatud kaks peidetud kihti, kus mõlemas on 32 LSTM sõlme. Mudel saavutab optimaalsed parameetrid 1100 epohhi korral. Sellisel juhul on mudeli kaoks 0,000881. Mudeli kadu on väiksem kui lihtsa mudeli oma, järelkult saadi parem mudel. Kuigi kadu saadi väiksem, siis kõige väiksem viga on nüüd $-0,2073$ ja suurim 0,1801. Kuna väiksema vea absoluutväärtus on suurem kui 0,2, siis võib eeldada, et leidub mingi aegrida, kus mudel ennustab, et väärtus jääb samaks, kuid tegelikult väärtus tõuseb. See eest on parandatud mudeli alumine ja ülemine kvartiil üksteisele lähemal. Mudeli üldine viga vähenes, kuid see eest leidub mõni väärtus, mille puhul mudel ei leia süstemaatilist muutust.



Joonis 5: Teise genereeritud andmete ennustus suuremal mudelil

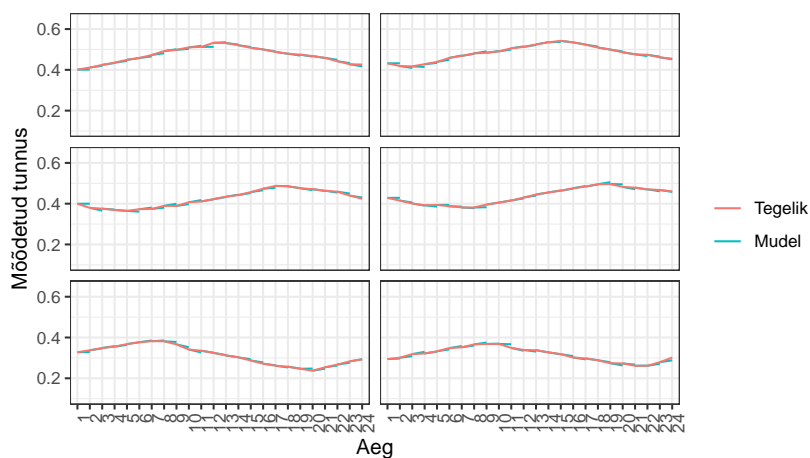
Jooniselt 5 on näha, et saadud mudel ennustab konkreetsete aegride puhul paremini kui eelnev mudel. On näha, et erinevalt eelnevast mudelist, siis enam ei ennusta mudel tõusu või langust seal,

kus seda ei ole. Alles jäänud mudeli mõõda ennustused on tingitud ainult andmete juhuslikkusest.

3.2.3 Kolmanda genereeritud andmestiku mudel

Kolmandale genereeritud andmestikule tehakse sama mudel, mis esimesele genereeritud andmestikule ja vaadatakse, kas mudel suudab leida seose kellaaaja ja mõõdetud tunnuse vahel. Ainuke erinevus esimese genereeritud andmestikuga on see, et sisendkihi suurus on üks. Optimaalsed parameetrid saadakse 3000 epohhi korral. Mudeli kadu on 0,000041. Kõige väiksem mudeli jääk on $-0,0288$ ja kõige suurem $0,0295$. Mõlemad on suuremad kui ajast tingitud kõikumine, kuid arvestades andmetes esinevat juhuslikkust, siis tegemist ei ole ka suurte jääkidega.

Vaadates joonist 6 on näha, et mudel ennustab täpselt. Mudel on hästi aru saanud üldiselt, kas järgmine väärtus on suurem või väiksem kui eelmine. Nagu näha, siis teise ajahetke ennustused on mudelil samad, mis eelneva ajahetke mõõtmiste tulemused, kuna peale ühte sisendit ei ole mudelil aimu, mis ajahetkega on tegemist ning kas mõõtmis tulemused peaksid langema või tõusma.

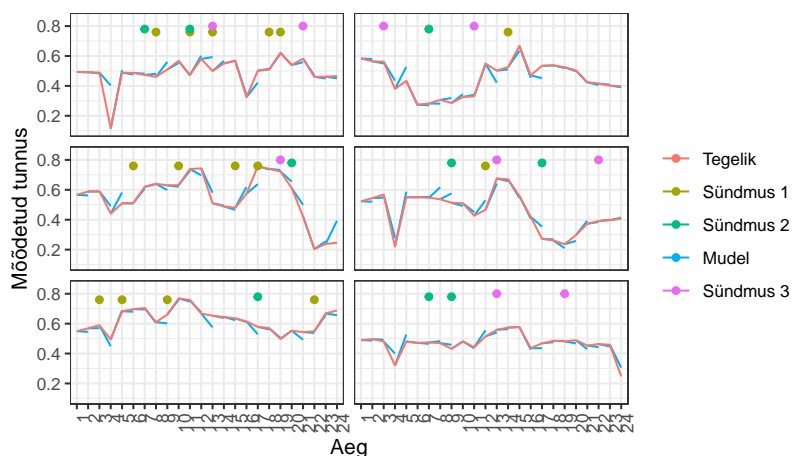


Joonis 6: Kolmanda genereeritud andmete ennustus

3.2.4 Neljanda genereeritud andmestiku mudel

Näha oli, et teise genereeritud andmestiku puhul läks täpsemate väärtuste saamiseks vaja suuremat mudelit, siis tehakse kohe neljanda genereeritud andmestiku peal suurem mudel nagu oli tehtud teise genereeritud andmestiku puhul. Erinevus varasema mudeliga on sisendkihi suuruses. Sisendkihi suurus antud mudeli korral on 4. Optimaalsed parameetrid leiab mudel 800 epohhi korral, mille puhul saadakse kaoks 0,002722. Nagu näha, siis saadud kadu on suurem kui eelnevate mudelite oma. Leides mudeli jäägid saadakse, et kõige väiksem jääk on 0,3639 ja kõige suurem 0.3765, Kuna jäägid on üsna suured, siis võib eeldada, et mudel ennustab mõnes kohas mööda ja ei suuda kindlat seost leida. Kui kõige suurem ja väiksem jääk välja jätta, siis mudeli ennustused on ikkagi üsna täpsed, kuna jääkide alumine kvartiil on 0,0154 ja ülemine 0,0083. Nagu ka teise mudeli puhul, siis jääkide arvutamisel on aegreast välja jäetud iga neljas tulemus. Arusaamaks täpsemalt kui hästi mudel ennustab, siis visualiseeritakse kuus mõõtmiste periood koos ennustustega.

Vaadates joonist 7 on näha, et üldiselt ennustab mudel hästi. Võrreldes varasemate tulemustega, siis antud mudel ei ole nii täpne nagu varasemad. On selgelt näha, et mööda ennustusi on natukene rohkem kui varasemate mudelite puhul.



Joonis 7: Neljanda genereeritud andmete ennustus

3.3 Järeldused

LSTM mudelid suutsid leida aegreast üles kõik eelnevad seosed ja nende põhjal ennustada täpselt tulevikuväärtusi. Mudelid said hästi aru, millist väärtust annab ette ennustada ja millist mitte nagu oli teise genereeritud andmestiku peal näha. Ühendatud seoste andmestiku puhul oli näha, et mudel ei olnud nii täpne nagu üksikute seoste puhul, kuid üldiselt oli mudel siiski täpne ja suutis teha enamik juhtudest täpseid ennustusi.

4 LSTM mudel meditsiini andmetel

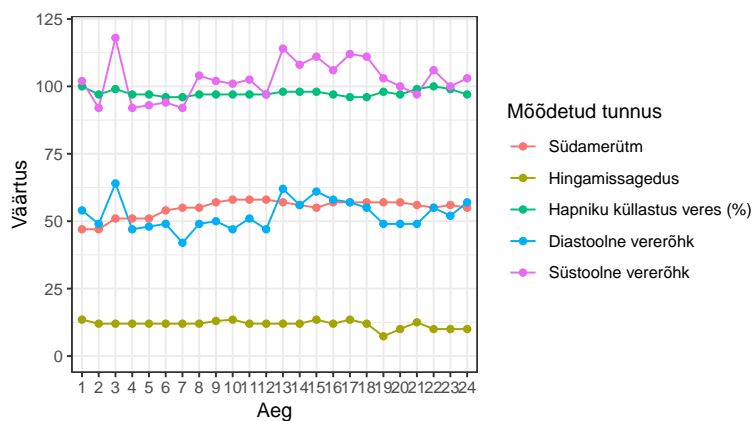
4.1 Andmestiku ülevaade

Andmed on pärit MIMIC-III andmestikust. MIMIC-III andmestikust on välja valitud kõik haiglakülastused, mille põhjus on olnud “hädaolukord” ja mille kestvus on olnud vähemalt 24 tundi. Samuti on alles jäänud ainult need külastused, mille puhul on iga tund mõõdetud järgnevat viite tunnust: südamerütm, hingamissagedus, hapniku küllastus veres, diastoolne vererõhk ja süstoolne vererõhk. Haiglakülastusi, mis täidavad eelpool mainitud tingimusi, on kokku 3230. Kõik nimetatud viis tunnust ka normeeritakse jagades läbi mõõdetud tunnused maksimaalse väärtusega. Südamerütmi puhul on maksimaalne väärtus 300, hingamissageduse korral 100, hapniku küllastus veres korral 100, diastoolse vererõhu korral 300 ja süstoolse vererõhu korral 400. Tunnused on olulised normeerida, et ühtlustada närvivõrgu sisendi väärtuseid. Kui närvivõrgu sisendid erinevad suuruse poolest liiga palju, siis järsima laskumise meetod ei pruugi koonduda globaalsesse miinimumi (Chollet ja Allaire, 2018).

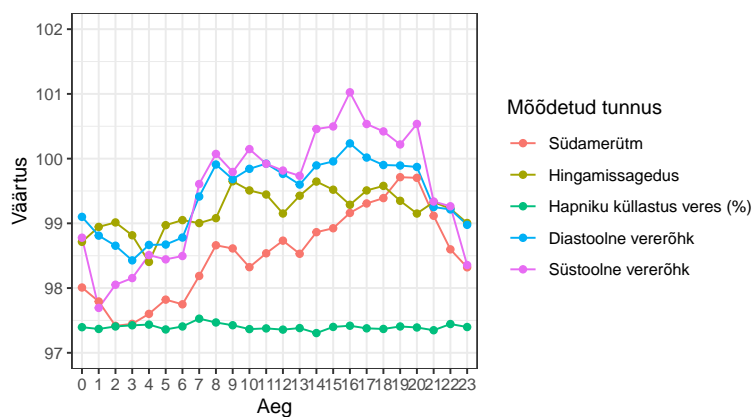
Lisaks on peale viie mõõdetava tunnuse teada veel patsiendi vanus, sugu ja mis ravimeid patsiendile konkreetsel tunnil manustati. Manustatud ravimite on alles jätud ainult need, mida on kokku manustatud kõigi patsientide peale kokku vähemalt 20 korda. Selliseid ravimeid on kokku 39. Iga allesjäänud ravim moodustab omaette üks null tunnuse, kus üks tähendab, et ravimit manustati ja null tähendab, et ravimit ei manustatud konkreetsel ajahetkel. Kokku on andmestikus 47 tunnust. Samuti normeeritakse vanus ja kellaaeg. Maksimaalne vanus MIMIC-III andmestikus on 90, ehk normeerimiseks tuleb vanus läbi jagada väärtusega 90. Kellaaja normeerimiseks jagatakse kõik kellaajad läbi väärtusega 23.

Peale vaadates ühe patsiendi mõõtmistele on silmaga raske öelda, et andmestikus oleks varasematel mõõtmistel seos tulevaste mõõtmis tulemustega (joonis 8). Samuti on raske näha ka kellaajalist seost, kuid seda oli raske näha ka neljanda genereeritud andmestiku puhul, kui muid mõjutavaid faktoreid on veel juures. Seepärast leitakse igal kellaajal tunnuste keskmine. Joonisel 9 on näha, et öösel on kõigi tunnuste peale hapniku küllastus veres mõõtmis tulemused madalamad kui päeval. Joonisel 9 on mõõdetud tunnuste keskmised nihutatud, et kõik mahuksid

samale skaalale. Igale südamerütmile keskmisele tulemusele on juurde liidetud väärtus 12, igale hingamissageduse mõõtmisele väärtus 80, igale diastoolse vererõhu keskmisele väärtus 40 ja igale süstoolse vererõhu keskmisele väärtus -21 . Hapniku küllastus veres keskmised on jäänud samaks. Juurde on liidetud antud väärtused, kuna suuremal skaalal ei tule südamerütmi korral kahe südamelöögi erinevus nii hästi esile.



Joonis 8: Ühe patsiendi mõõtmised 24 tunni jooksul



Joonis 9: Ühe patsiendi mõõtmised 24 tunni jooksul

4.2 Andmete jaotamine

Mudelite tegemiseks on andmestik viidud kolmemõõtmelisele kujule. Esimene mõõde on haiglakülastus, teine mõõde on aeg tundides ja kolmas mõõde tunnused. Andmestiku mõõtmed on (3230, 24, 47). Mudeli treenimiseks ja testimiseks jaotatakse andmestik kaheks: treeningandmestik ja valideerimisandmestik. Külastused jaotatakse treeningandmestikku ja valideerimisandmestikku juhuslikult. Iga külastuse tõenäosus sattuda treeningandmestikku on 0,8 ja valideerimisandmestikku 0,2. Peale juhuslikku jaotamist on 3230st külastusest sattunud treeningandmestikku 2606 külastust ja valideerimisandmestikku 624 külastust. Nii treeningandmestikust kui valideerimisandmestikust moodustatakse sisendandmestik ja tulemusandmestik. Nagu genereeritud andmete puhul, siis sisendandmestiku moodustavad esimesed 23 mõõtmist ja tulemusandmestiku moodustavad viimased 23 mõõtmist.

4.3 Mudelite ehitus ja treenimine

Teise genereeritud andmestiku peal oli näha, et ennustuste täpsus sõltub ka mudeli ehitusest. Saamaks teada optimaalset mudeli ehitust päris andmete peal, tehakse viis mudelit. Mudelid erinevad üksteisest kihi arvude ja kihis olevate sõlmede koguse poolest. Samuti on kahele mudelile lisatud väljajätumeedod. Kõigi mudelite sisendkihid on samad. Sisendkihi suurus on 47.

Esimene mudel on kõige lihtsam ning koosneb kolmest kihist. Teises kihis ehk peidetud kihis on 32 LSTM sõlme. Viimane kiht ehk väljundkiht koosneb viiest lihtsast sõlmest, kuna mudeliga tahetakse saada ka viite erinevat tulemust.

Teine mudel on võrreldes esimesega sügavam. Kokku on mudelis viis kihti. Sisendkiht ja väljundkiht on samasugused nagu esimeses mudelis. Esimeses peidetud kihis on 32 LSTM sõlme, teises peidetud kihis on 64 LSTM sõlme ja kolmandas 32 LSTM sõlme.

Kolmas mudel on teise mudeliga sarnane. See eest on lisatud peale esimest ja teist peidetud kihti väljajätumeedod. Mõlemas väljajätumeedodis on väljajätumäär 0,5.

Neljas mudel on veel suurem kui teine mudel. Mudelisse on juurde lisatud veel üks kiht ehk kokku on mudelis kuus kihti. Samuti on esimeses peidetud kihis 32 LSTM sõlme asemel 64 LSTM sõlme.

Teises peidetud kihis on 128, kolmandas peidetud kihis 64 ja neljandas peidetud kihis 32 LSTM sõlme. Väljundkiht on sama, mis eelnevatel mudelitel.

Viies mudel on sarnane neljanda mudeliga, aga juurde on lisatud väljajätumeedid. Väljajätumeedod on lisatud peale esimest, teist ja kolmandat peidetud kihti. Igas väljajätumeedis on väljajätumäär 0,5.

Mudelite treenimist on täpsemalt näha lisa 3 all. Esimese mudeli korral saadakse optimaalsed parameetrid 300 epohhi korral. Sellisel juhul saadakse kaoks 0,000914. Teise mudeli korral saadakse optimaalsed parameetrid samuti 300 epohhi korral. Teise mudeli korral on kadu 0,000914. Ümardamise tõttu ei tule see selgelt välja, aga mudeli kadu on võrreldes esimese mudeliga natuke suurem. Kolmanda mudeli korral saadakse optimaalsed parameetrid 700 epohhi korral kaoga 0,000926. Neljanda mudeli puhul saadakse optimaalsed parameetrid 250 epohhi korraga. Neljanda mudeli puhul on kaoks 0,000925. Viienda mudeli korral saadakse optimaalsed parameetrid 300 epohhi korral. Kaoks tuleb sellisel juhul 0,000975.

Tabel 1: Mudelite epohhile kulumise aeg

Mudel	Ühe epohhi aeg
Esimene mudel	~ 1s
Teine mudel	~ 4s
Kolmas mudel	~ 4s
Neljas mudel	~ 5s
Viies mudel	~ 5s

Nagu näha, siis üldiselt saavutasid kõik mudelid optimaalsed parameetrid enam vähem 300 epohhi korral. Ainult kolmanda mudeli korral läks teistest mudelitest võrreldes märksa rohkem epohhe vaja. See eest oli ka oodatav, et kolmanda mudeli treenimiseks läheb kauem kui teise mudeli treenimiseks ja viienda mudeli treenimiseks läheb kauem kui neljanda mudeli treenimiseks, kuna mõlemad mudelid olid saadud eelnevast mudelist väljajätumeedodite lisamisel. Kuna kõige lihtsam mudel andis kõige väiksema kao ja sellele kulus kõige vähem epohhe, siis oli ka selle tegemine kõige optimaalsem, kuna vaadates tabel 1 on näha, et ühele epohhile kulub selle korral

kõige vähem aega. Samuti on näha, et väljajätumeedod ei suurenda epohhile kuluvat aega. Tablis 1 epohhidele kuluv aeg saavutati, kui mudelite arvutamisel kasutati nelja Nvidia Tesla V100.

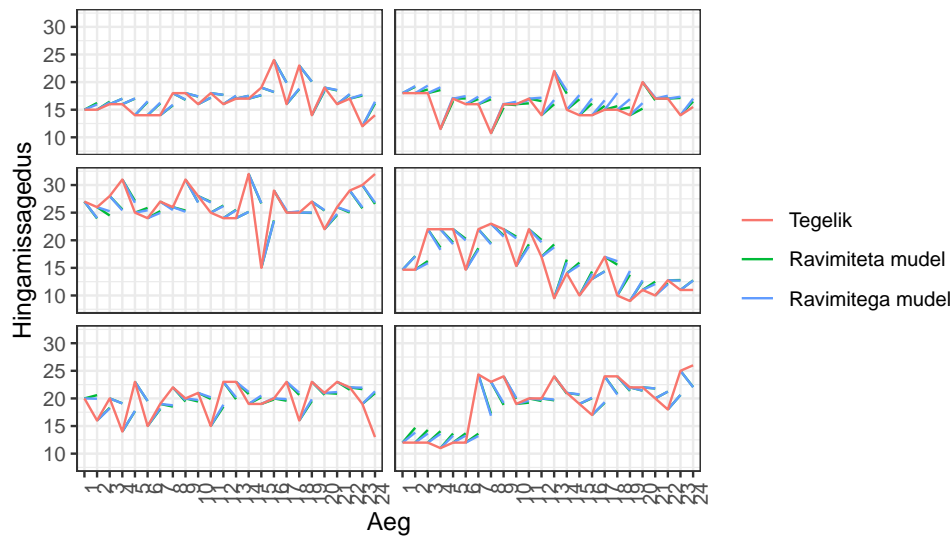
Kuna lihtne mudel andis kõige väiksema kao, siis vaadatakse, kas veel lihtsam mudel annab parema tulemuse. Selleks tehakse uus mudel, mis on sarnane esimese mudeliga, aga peidetud kihis on sõlmede arvu vähendatud 32 sõlme pealt 16 sõlme peale. Optimaalsed parameetrid saavutab lihtsam mudel 500 epohhi korral kaoga 0,000912. Nagu näha, siis kadu saadi veel väiksem kui esimese mudeli korral. See eest kulus optimaalsete parameetrite leidmiseks 200 epohhi rohkem ning ühele epohhile kulub endiselt sama kaua aega kui esimese mudeli korral.

Kuna lihtne mudel töötas kõige paremini, siis vaadatakse, kas mudelit veel lihtsustades saadakse parem tulemus. Võetakse eelmise mudeliga sama ehitus, aga mudeli sisendist jäetakse ära ravimid. Sellisel juhul on mudelil 47 sisendi asemel kaheksa sisendit ehk sisendkihi suurus on 47 asemel kaheksa. Mudel saavutab optimaalsed parameetrid 550 epohhi korral. Mudeli kaoks saadakse sellisel juhul 0,000904. Nagu näha, siis optimaalsete parameetrite leidmiseks kulub 50 epohhi rohkem. Saadud mudeli kadu on veel madalam kui koos ravimitega saadud mudeli korral. Võrreldes genereeritud andmetega ei ole päris andmestikus ravimi mõju igale patsiendile sama ja päris ravimite koosmõjud, erinevalt genereeritud andmestiku omast, võivad tekitada teistsuguseid koosmõjusid, mida on keerulisem ette ennustada.

4.4 Mudelite tulemused ja võrdlus

Tulemuste analüüsimiseks ja ravitega saadud mudeli ja ravimiteta saadud mudeli võrdlemiseks, võetakse parim ravimitega saadud mudel ja ravimiteta tehtud mudel. Parimal ravimitega mudelil on sama ehitus, mis ravimiteta mudelil. Mõlemad mudelil koosnevad kahest kihist, kus sisendkihis on 16 LSTM sõlme ja väljundkihis 5 lihtsat sõlme. Nägemaks kuidas mõlemad mudelid ennustavad, võetakse juhuslikult kuus patsienti ja visualiseeritakse nende mõõtmiste tulemused ning mudelite ennustused. Mõõtmiste tulemuste ja mudelite ennustuste joonistel on mõõtmised teisendatud tagasi algsele skaalale ehk normeeritud väärtused on läbi korrutatud mõõdetud tunnuse maksimaalse väärtusega.

Esimene mõõdetud tunnus oli südamerütm. Vaadates joonist 10 on näha, et kui südamerütm on

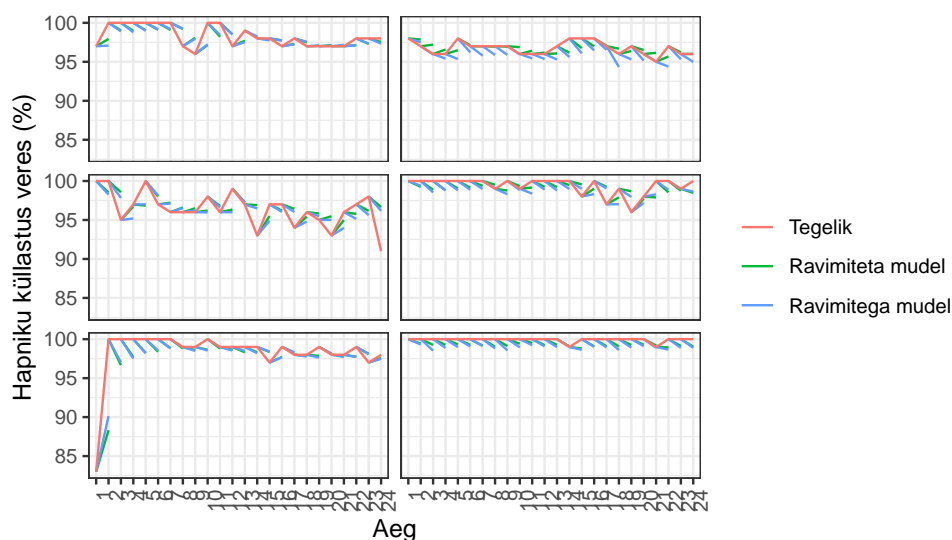


Joonis 11: Patsientide hingamissageduse mõõtmised koos ennustustega

See eest kui peale suuremat tõusu väärtus ei lange nagu alumisel paremal joonisel, siis mudel ennustab paljuga mööda.

Leides mõlema mudeli ruutkeskmise vea hingamissageduse korral saadakse, et ravimitega mudeli korral on see 0,001616 ja ravimiteta mudeli korral 0,001599. Nagu näha, siis antud kaod on suuremad kui mudeli üldine kadu. Viies kaod tagasi algsele skaalale saadakse, et ravimitega mudeli korral on ruutkeskmise viga 16,1622 ja ravimiteta mudeli korral 15,9918. Keskmiselt on mõlema mudeli eksimuse ruut 16 hingetõmmet.

Võrreldes teiste tunnustega, siis hapniku küllastus veres on ainuke tunnus, mida on mõõdetud protsentides. Samuti on see ainuke tunnus, mille puhul on palju mõõtmisi maksimaalse väärtusega. Nagu jooniselt 12 on võimalik näha, siis hapniku küllastus veres on olnud üldiselt stabiilsem tunnus. See eest on näha, et kumbki mudel ei ole suuda ette ennustada tunnuse muutust. On näha, et kui tunnus on täiesti stabiilne nagu alumisel paremal poolsel joonisel, siis mudel ennustab igal ajahetkel langust. Üldiselt on tunnus olnud stabiilne siis, kui hapniku küllastus veres mõõtmised on olnud 100% peal, mis tõttu võib olla mudelil probleem ennustada järjest maksimaalset väärtust.

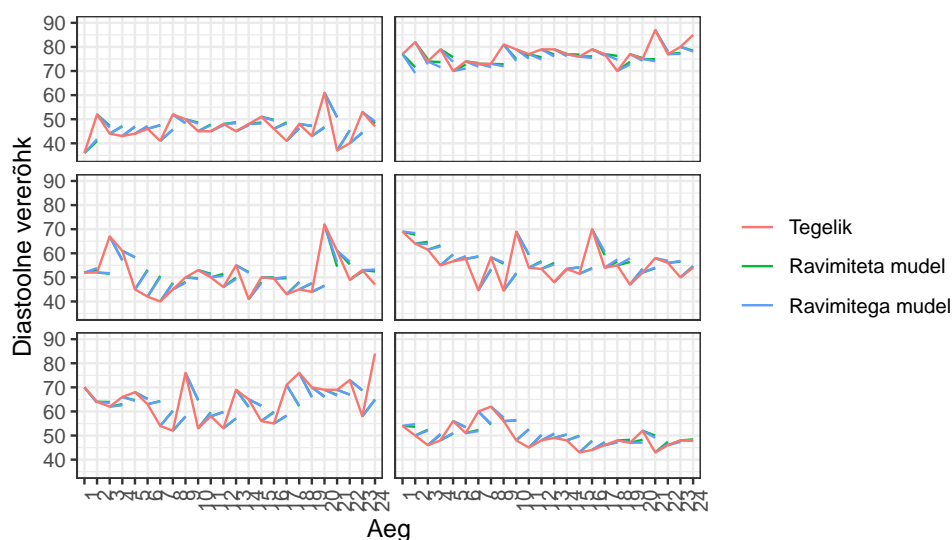


Joonis 12: Patsientide hapniku küllastuse veres mõõtmised koos ennustustega

Vaadates mudelite kadusi hapniku küllastus veres korral, siis ravimitega mudeli korral on mudeli ruutkeskmine viga 0,000572 ja ravimiteta mudeli korral 0,000544. Nagu näha, siis mõlemad kaod on väiksemad kui mõlema mudeliga saadud üldine kadu. Viies kaod tagasi algsele skaalale saadakse, et ravimitega mudeli korral on ruutkeskmine viga 5,7191 ja ravimiteta mudeli korral 5,4394. Keskmiselt on mõlema mudeli eksimuse ruut 5,5%.

Vaadates joonist 13 on näha, et diastoolne vererõhk on kõikunud palju, mistõttu ka ennustused erinevad tegelikkusest palju. See eest kui vaadata ennustusi on mõlemad mudelid mõnel juhul ennem suurt langust või tõusu suutnud seda ka ette ennustada.

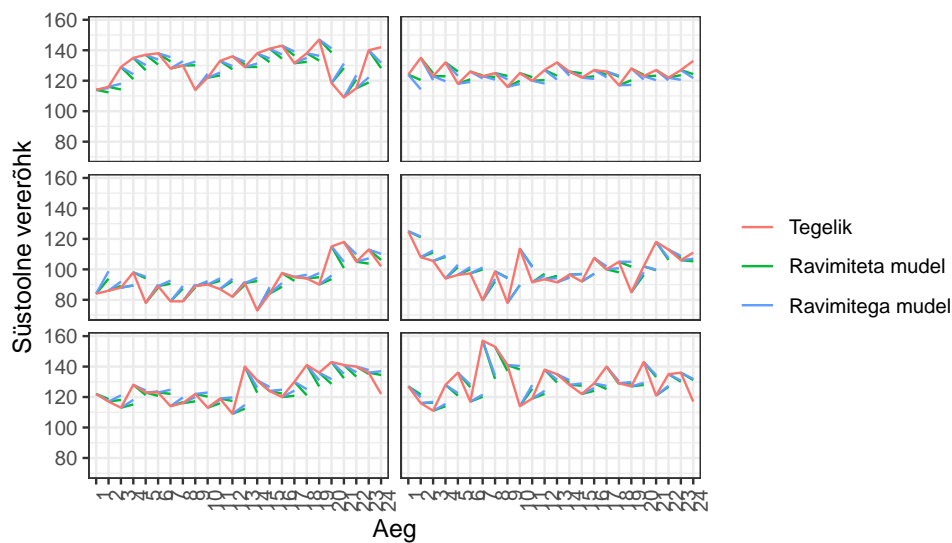
Vaadates ruutkeskmise viga diastoolse vererõhu korral saadakse, et ravitega mudeli korral on see 0,000750 ja ravimiteta mudeli korral 0,000743. Mõlemad on madalamad kui mudelitega saadud üldine ruutkeskmine viga. Teisendades mõlemad tagasi algsele skaale saadakse, et ravimiga mudeli korral on ruutkeskmine viga 67,4656 ja ravimiteta juhul 66,9078. Keskmiselt on mudeli ennustuse ja tegelikkuse vahe ruut 67 mmHg.



Joonis 13: Patsientide diastoolse vererõhu mõõtmised koos ennustustega

Jooniselt 14 on näha, et samuti ka süstoolse vererõhu korral, kui mõõdetud tunnus ei ole stabiilne, siis mudelid ennustavad kehvasti. Veel on näha, et nagu ka varasemate tunnuste korral, siis peale stabiilset ajaperioodi ei suuda mudel ette ennustada suuremat langust või tõusu.

Seda, et mõlemad mudelid ennustavad kehvasti süstoolse vererõhu korral, kinnitavad ka mõlema mudeli ruutkeskmised vead. Ravimitega mudeli korral on ruutkeskmine viga 0,001085 ja ravimiteta mudeli korral 0,001091. Samuti on näha veel, et tegemist on ainukese tunnusega, kus ravimiteta mudeli korral on ruutkeskmine viga suurem kui ravimitega mudeli korral. Mõlemad vead on suuremad kui üldine mudeli kadu. Teisendades mõlemad ruutkeskmised vead tagasi algsele skaalale saadakse, et ravimitega mudeli ruutkeskmine viga on 173.5315 ja ravimiteta mudeli korral 174.4981. Keskmiselt on mudeli ennustuse ja tegelikkuse vahe ruut 174 mmHg.



Joonis 14: Patsientide süstoolse vererõhu mõõtmised koos ennustustega

4.5 Mudelite järeldused

Üldiselt oli näha, et mõlemad valitud mudelid ennustasid samu tulemusi ehk ei olnud vahet, kas mudelisse kaasata ravimid või mitte. Normeeritud väärtuste korral oli näha, et kõige suurema kao andis tunnus hingamissagedus ja kõige madalama andsid südamerütm ja hapniku küllastus veres. Kaks kõige madalama kaoga tunnust olid ka kõige stabiilsemad.

Sellele, et mõnel juhul mudelid töötasid kehvasti, võib olla mitu põhjust. Esiteks võib olla liiga vähe andmeid. Mõned seosed võisid olla antud valimis vähe esindatud ja ei pruukinud välja tulla. Suurema treeningandmestiku korral võivad ajas esinevad seosed tulla paremini välja. Teiseks, üks tund on pikk aeg ja selle jooksul võivad mõõdetud näitajad muutuda palju, mistõttu on liiga palju juhuslikust järgneva tunni mõõtmises. Andmestiku korral, kus on mõõdetud tunnuseid tihedamini, näiteks iga kümne minuti tagant, võivad esineda tugevamad seosed, mida on võimalik ka paremini ette ennustada. Kui näiteks võtta teine genereeritud andmestik ja sealt jätta alles ainult iga teine väärtus, siis on andmete vahel väga nõrk seos ning võimatu on ette ennustada, kas mõõdetud tunnus langeb, tõuseb või jääb samaks. MIMIC-III andmestikus on olemas pat-

siente, kellel on ka antud tunnuseid tihedamini mõõdetud, kuid neid patsiente on liiga vähe ja mõõtmisperioodi pikkus on neil patsientidel olnud liiga lühike.

Kokkuvõtte

Töö eesmärk oli välja selgitada, kas rekurrentsed närvivõrgud suudavad leida üles aegreast seoseid ja nende põhjal tuleviku väärtusi ennustada. Genereeritud andmete peal oli näha, et tehtud mudelid suutsid leida aegreast üles sündmustest põhjustatud mõju, süstemaatilise mõju ja tsüklilise mõju ning nende põhjal teha täpseid tuleviku väärtuste ennustusi. Samuti suutsid tehtud närvivõrgu mudelid leida kõik seosed, kuid need olid ühes andmestikus koos.

MeditSiini andmete peal, kus oli aluseks võetud andmestik MIMIC-III, prooviti leida seoseid tunnuste südamerütm, hingamissagedus, hapniku küllastus veres, diastoolne vererõhk ja süstoolne vererõhk korral. Meditsiini andmete peal ei töötanud LSTM mudelid nii hästi nagu genereeritud andmete peal, mis oli ka ootuspärane. Andmete puhul küll leidsid mudelid üles mingid seosed ja suutsid selle põhjal ka tuleviku väärtusi natukene ette ennustada. Saamaks paremaid tulemusi päris andmete peal, tuleb suurendada näiteks valimi mahtu või suurendada mõõtmisagedust, kuna ühe tunni mõõdudes võib tekkida liiga palju juhuslikust järgneva mõõtmis tulemuse ja eelnevate mõõtmis tulemuste vahel.

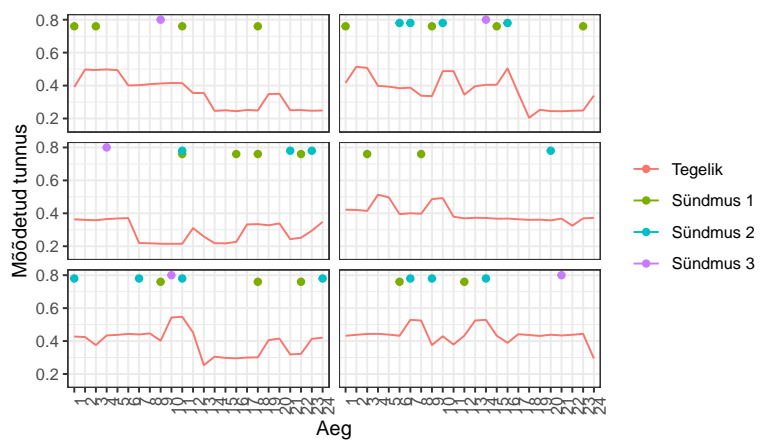
Veel oli näha, et igale olukorrale sai rakendada sama mudeli ehitust, milleks oli lihtne kolme kihiline mudel. Mudel töötas igal genereeritud andmestikul ja andis ka meditsiini andmetel parima tulemuse. Samuti oli näha meditsiini andmestikul mudeleid treenides, et lihtsat mudelit on kiirem treenida, mistõttu oli ka selle tegemine ajaliselt efektiivsem.

Kasutatud kirjandus

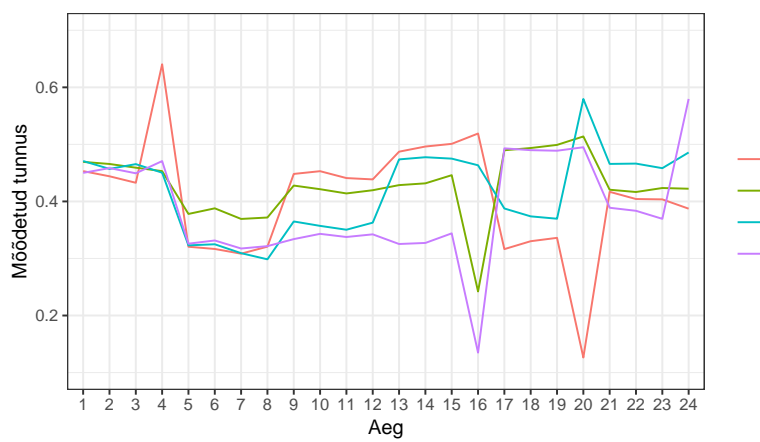
- Abrahamson, J., 2018. *Choosing loss functions*. Kasutatud 07.05.2020, <https://simplicial.io/2019/04/26/choosing-loss-functions/>
- Callan, R., 2003. *Artificial Intelligence*. Basingstoke: Palgrave Macmillan
- Chollet, F. ja Allaire, J. J., 2018. *Deep Learning with R*. Shelter Island: Manning
- Kathuria, A., 2018. *Intro to optimization in deep learning: Momentum, RMSProp and Adam*. Kasutatud 07.05.2020, <https://blog.paperspace.com/intro-to-optimization-momentum-rmsprop-adam/>
- Kingma, D. P. ja Ba, J., 2017. *Adam: A Method for Stochastic Optimization*. arXiv:1412.6980
- Luschi, C. & Masters, D., 2018. *Revisiting Small Batch Training for Deep Neural Networks*. arXiv:1804.07612
- Murphy, K. P., 2012. *Machine Learning: A Probabilistic Perspective*. Cambridge: MIT Press.
- Nilsson, N. J., 1998. *Artificial Intelligence: A New Synthesis*. San Francisco: Morgan Kaufmann Publishers
- Olah, C., 2015. *Understanding LSTM Networks*. Kasutatud 07.05.2020, <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- Rizwan, M., 2018. *Gradient Descent with Momentum*, Kasutatud 07.07.2020, <https://engmrk.com/gradient-descent-with-momentum/>
- Russel, S. J. & Norvig, P., 2016. *Artificial Intelligence: A Modern Approach*. 3rd ed. Harlow: Pearson.

Lisad

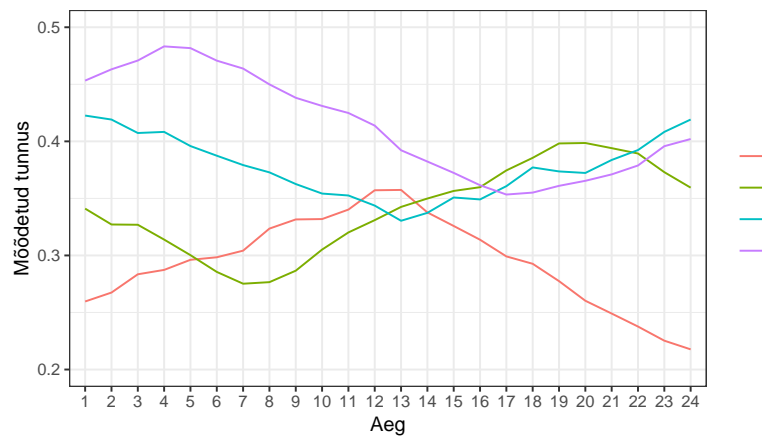
Lisa 1, Genereeritud andmed



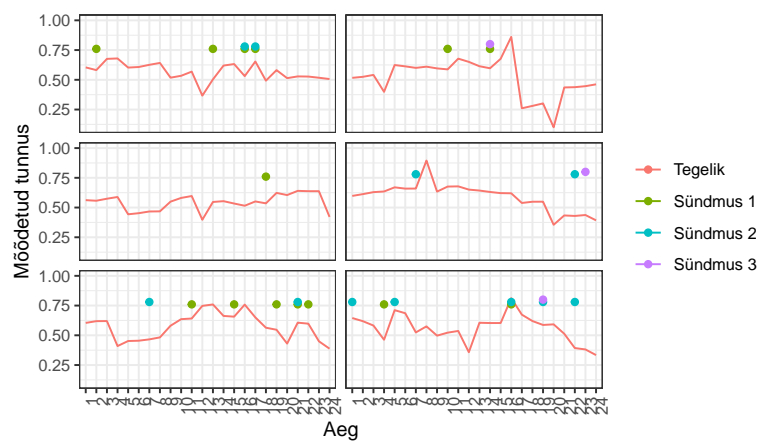
Joonis 15: Esimene genereeritud andmestik



Joonis 16: Teine genereeritud andmestik

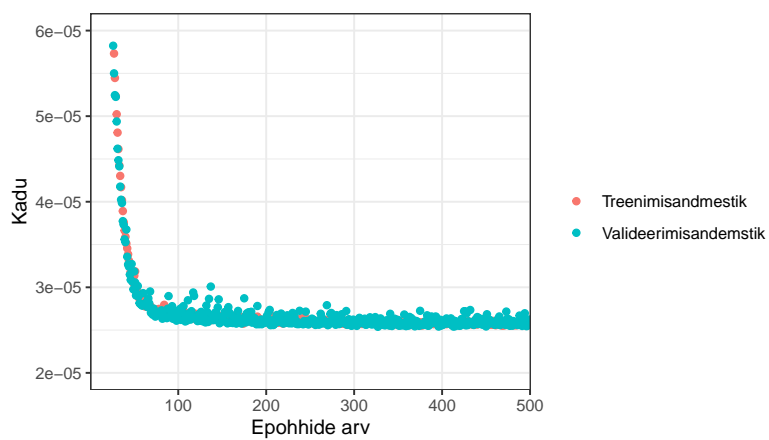


Joonis 17: Kolmas genereeritud andmestik

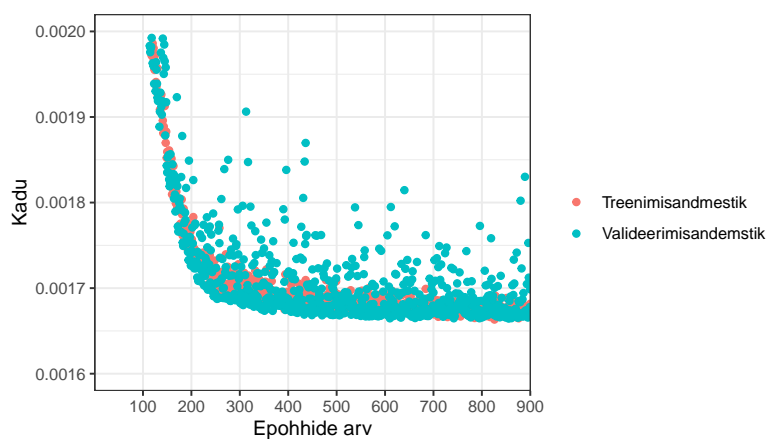


Joonis 18: Neljas genereeritud andmestik

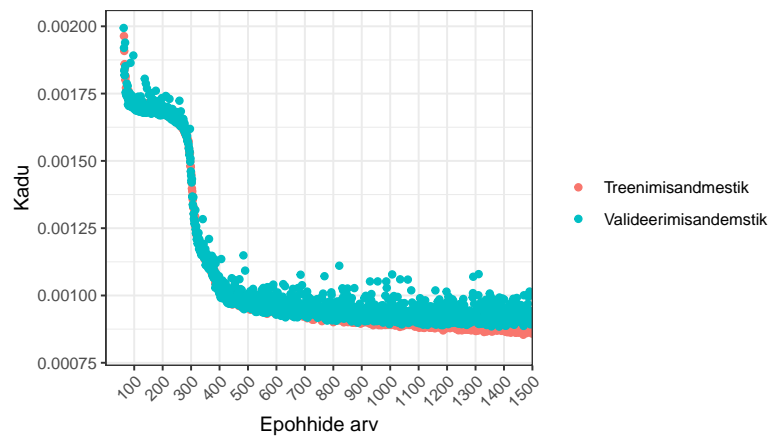
Lisa 2, Mudelite treenimine genereeritud andmetel



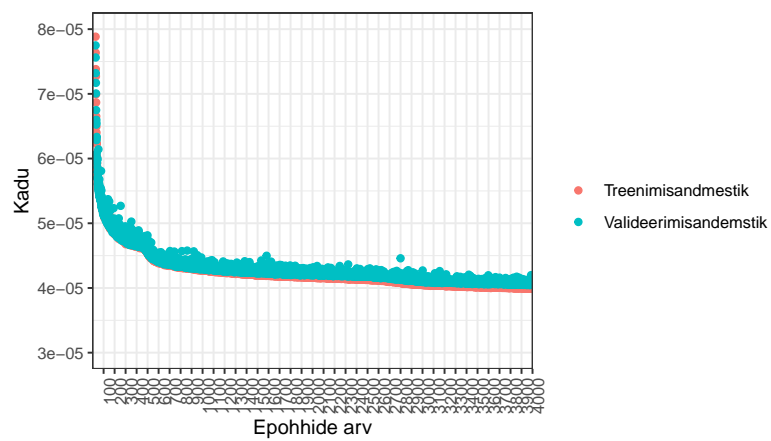
Joonis 19: Esimese genereeritud andmestiku mudeli treenimine



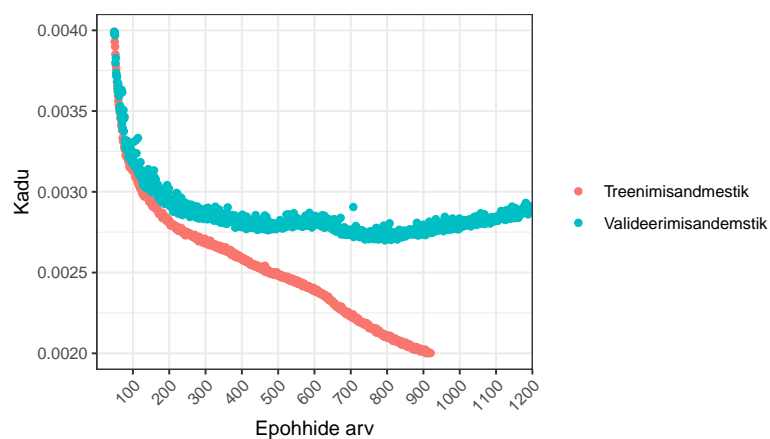
Joonis 20: Teise genereeritud andmestiku mudeli treenimine



Joonis 21: Teise genereeritud andmestiku suurema mudeli treenimine

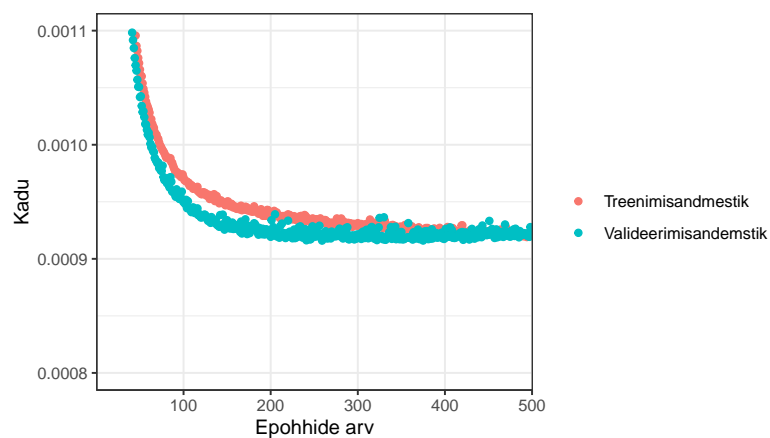


Joonis 22: Kolmanda genereeritud andmestiku mudeli treenimine

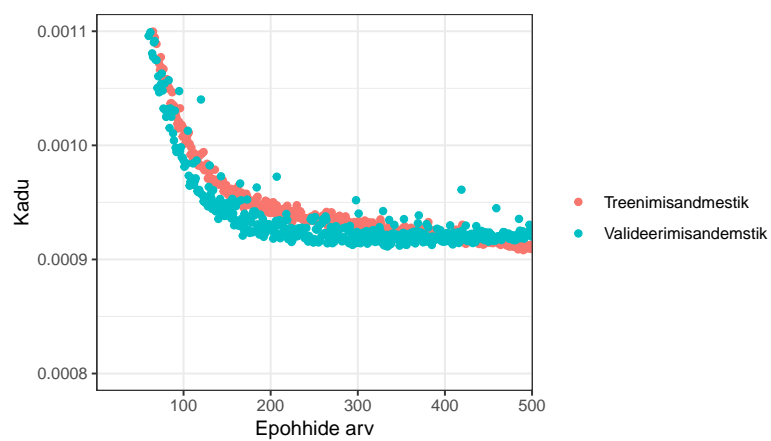


Joonis 23: Neljanda genereeritud andmestiku mudeli treenimine

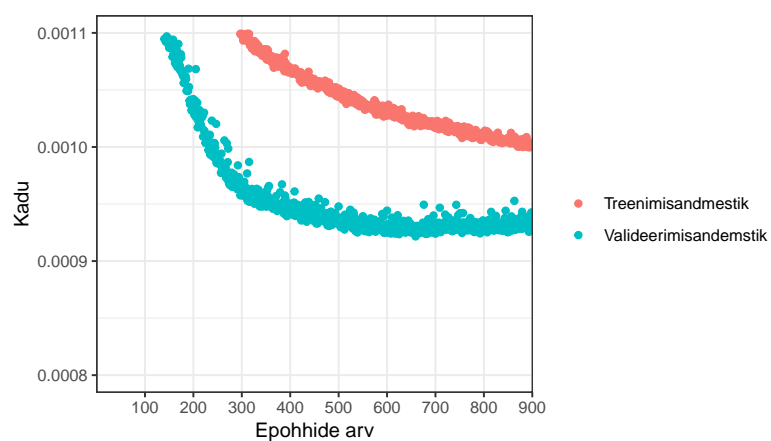
Lisa 3, Mudelite treenimine meditsiini andmetel



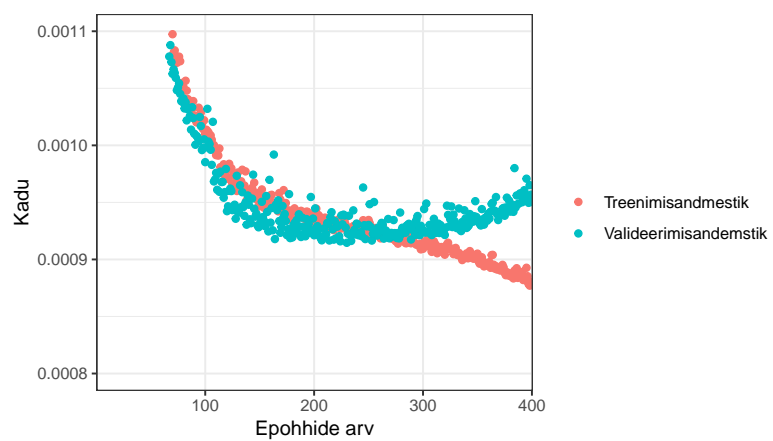
Joonis 24: Esimese mudeli treenimine



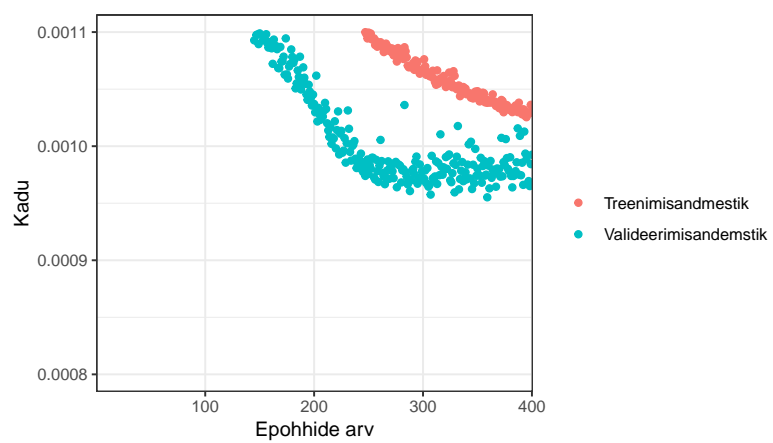
Joonis 25: Teise mudeli treenimine



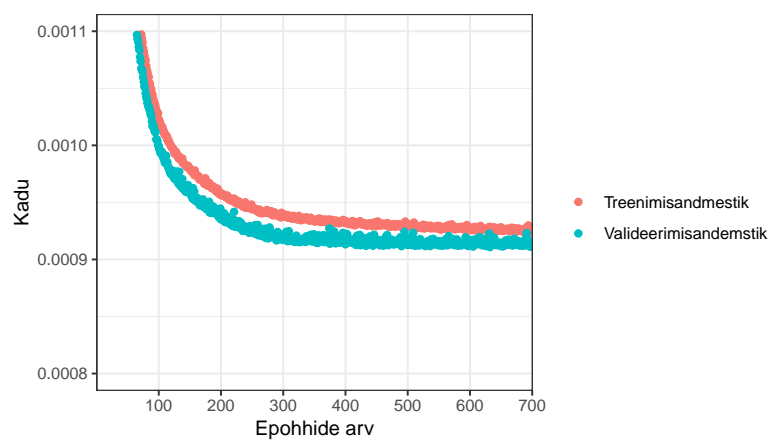
Joonis 26: Kolmanda mudeli treenimine



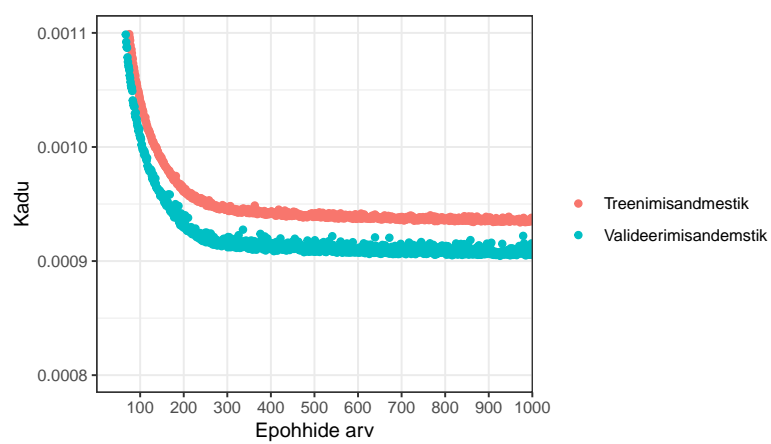
Joonis 27: Neljanda mudeli treenimine



Joonis 28: Viienda mudeli treenimine



Joonis 29: Lihtsustatud mudeli treenimine



Joonis 30: Lihtsustatud ravimiteta mudeli treenimine

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Mihkel Lepson

1. Annan Tartu Ülikoolile tasuta load (lihtlitsents) minu loodud teose “Rekurrentsed Närvi-võrgud Aegridade Uurimiseks”, mille juhendaja on Raivo Kolde reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.
2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Mihkel Lepson

18.05.2020